

**Foundations and Trends® in Electric Energy
Systems**

Circuit-Based Electromagnetic Transient Simulation

Suggested Citation: Amritanshu Pandey (2023), “Circuit-Based Electromagnetic Transient Simulation”, Foundations and Trends® in Electric Energy Systems: Vol. 6, No. 2, pp 83–118. DOI: 10.1561/31000000038.

Amritanshu Pandey
University of Vermont
Amritanshu.Pandey@uvm.edu

This article may be used only for the purpose of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval.

now
the essence of knowledge
Boston — Delft

Contents

Introduction	84
1 Equivalent Circuit Approach for EMT Simulations	86
1.1 Wye-Connected RL Load	88
1.2 Induction Motor Load	97
Conclusion and Resources for a Deeper Dive	113
Notation	114
Acknowledgments	116
References	117

Circuit-Based Electromagnetic Transient Simulation

Amritanshu Pandey

University of Vermont, USA; Amritanshu.Pandey@uvm.edu

ABSTRACT

The growing penetration of inverter-based resources and associated controls necessitates system-wide electromagnetic transient (EMT) analyses. EMT tools and methods today were not designed for the scale of these analyses. In light of the emerging need, there is a great deal of interest in developing new techniques for fast and accurate EMT simulations for large power grids; the foundations of which will be built on current tools and methods. However, we find that educational texts covering the fundamentals and inner workings of current EMT tools are limited. As such, there is a lack of introductory material for students and professionals interested in researching the field. To that end, in this tutorial, we introduce the principles of EMT analyses from the circuit-theoretic viewpoint, mimicking how time-domain analyses are performed in circuit simulation tools like SPICE and Cadence. We perform EMT simulations for two examples, one linear and one nonlinear, including induction motor (IM) from the first principles. We anticipate that by the end of this monograph, the readers will have a basic understanding of how power grid EMT tools work.

Introduction

This monograph is a tutorial to help students and working professionals learn the inner workings of tools that perform electromagnetic transient (EMT) simulations. The tutorial uses a circuit-theory based approach to EMT simulation. The tutorial is introductory in that it shows the workings through two simple examples and expects the reader to build expertise through more detailed readings. Nonetheless, this monograph is an excellent first step for anyone interested in the subject, especially those interested in building their own tools.

Before we dive into EMT models and simulation, it is pertinent to discuss and differentiate between the two types of commonly performed time-domain simulations in the power community:

- **Transient Stability (TS):** Performs time-domain simulations on balanced networks (only positive sequence components). It does not include network impedance transient response and models them moving from one steady state to another instantaneously. Therefore transient stability frameworks model the network constraints with algebraic equations and only model the injection components and controls with differential algebraic equations. TS is also sometimes called root-mean-squared (RMS) transient analysis. These have been traditionally the workhorse for system-wide time-domain power grid analysis.

- **Electromagnetic Transient Simulation:** Works exactly as circuit-simulation tools (for instance, SPICE and CADENCE). It models all three phases, including the transient response for the network impedance components. The industry demand for large-scale EMTs is growing rapidly due to the recent events involving power electronics devices.

This monograph focuses on electromagnetic transient simulation. These are becoming increasingly common (with significant active research) due to the introduction of inverter-based resources on the grid. The monograph provides a step-by-step tutorial on performing an EMT simulation for simple networks from the first principles. It uses concepts from circuit simulation, Newton-Raphson, and numerical integration with difference methods. Good references to brush up on these topics are [9], [4], and [1]. The monograph does not discuss modeling controls in EMT but provides sufficient background such that it can be tackled as the next step.

Furthermore, this monograph uses modified nodal analysis (MNA) and sometimes loop analysis to encapsulate network physics. Other alternate approaches like Tree Link Analysis (TLA) and Sparse Tableau Analysis (STA) [9] exist but are not covered in this monograph. In naive nodal analysis, the currents at each node are summed to zero (i.e., Kirchhoff's current law) to satisfy network physics. However, the method fails when the network includes voltage sources, as the current through the voltage source is not implicitly known. Therefore, to include voltage sources, the modified version adds one additional constraint per voltage source to the set of nodal equations. These additional constraints give us the currents through the voltage sources as new variables. The term *modified* in MNA refers to this modification. Finally, the readers must note that anytime in this monograph when they come across the following terms: system matrix, Y matrix, solution matrix, or simple nodal matrix, these all refer to the same thing: a set of nodal or loop equations (or sometimes a mix) for the linearized network that is being evaluated at a given time-step t .

1

Equivalent Circuit Approach for EMT Simulations

We will use an equivalent circuit framework [7], [8] for power grids to develop EMT models for two simple toy networks. These toy examples are based on a simple 2-bus three-phase power network with different load configurations. We consider the following load scenarios:

- A linear wye-connected series RL load (see Figure 1.3).
- A non-linear induction motor load (see Figure 1.11).

Note: For both examples, we assume that the load is connected in a wye configuration and all the neutrals in the network are grounded.

The illustration of the two-bus power network with a generic load model is shown in Figure 1.1. It includes power system verbiage such as infinite bus and wye-connected load. For the moment, the infinite bus (at nodes a' , b' , c') can be considered the power source, and the load bus (at nodes a , b , c) can be considered the power sink.

The illustration in Figure 1.1 is a power system representation. To perform EMT simulation with an equivalent circuit framework, as a first step, we represent any power grid-specific elements as electric circuit elements. The network in Figure 1.1 has three components: (i) an infinite bus, (ii) a transmission line (a very simplified model), and (iii) a load.

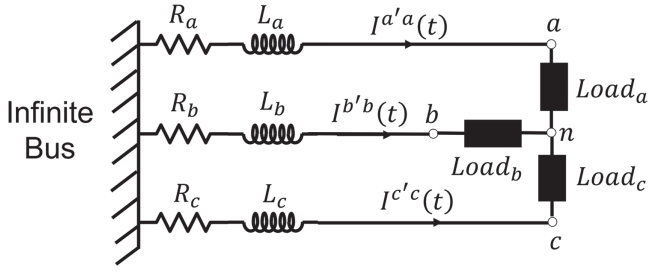


Figure 1.1: Power system representation of the 2-bus power network.

Two of these three components (infinite bus and transmission line) are shared across the two examples. Therefore, we begin with them. The infinite bus is trivial to model as a circuit element. It is simply an ideal voltage source with a known magnitude and phase angle. Further, the frequency of this source is assumed to be either 60 Hz or 50 Hz depending on the system. For a three-phase network, the infinite bus is modeled by a set of three independent voltage sources (offset by 120°) connected in a wye formation. The network with an infinite bus replaced with circuit elements is shown in Figure 1.2. The transmission line model with R and L values already represents a set of circuit elements and does not require translation. Note that this model of the transmission line is greatly simplified to improve the readability of this tutorial. A more detailed model (see [3]), at a minimum, should include shunt capacitances and mutual impedances. Longer lines should use lumped models. Still, the readers will learn to model mutual impedances while developing the EMT model for an induction motor.

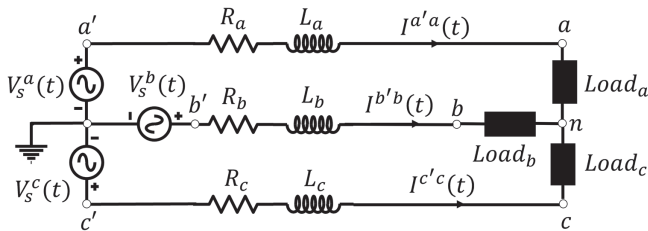


Figure 1.2: Power system representation of the 2-bus power network.

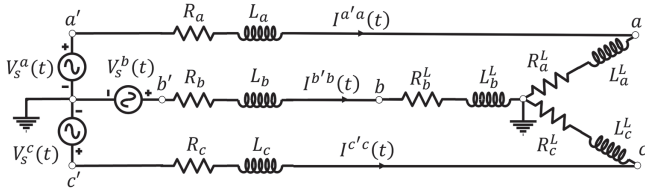


Figure 1.3: Power system representation of the 2-bus power network.

The last step is to convert the load component in Figure 1.2 into an equivalent circuit before getting started with EMT-related steps. Loads in power systems correspond to various devices ranging from large induction motors to resistive heating loads. We will translate the two load models into equivalent circuits in their respective subsections.

1.1 Wye-Connected RL Load

The first example represents the load via a wye-connected series RL impedance. With series RL impedance as the load component, the equivalent circuit representation of the 2-bus network is trivial and is shown in Figure 1.3. Simple observation confirms that all elements in this equivalent circuit are linear; therefore, to obtain the transient response, we only need to *recursively* solve the network as a function of time. *Iterative* solves with Newton-Raphson (NR) are only necessary in the presence of nonlinear components. We will learn about that in the induction motor (IM) example.

In current EMT tools (for instance, EMTP-RV or PSCAD), a sequence of steps that are followed to obtain the time-domain response of a linear power network are as follows:

- (1) First, ordinary differential equations (ODEs) corresponding to the sum of currents at each node in the network are defined.
- (2) Second, numerical integration approximation (such as trapezoidal or backward Euler) is applied to these ODE equations to convert them from differential algebraic equation (DAE) form to purely algebraic equation (AE) form.

- (3) Third, if the network is linear, the final solution for each time instance is obtained via a recursive linear solve of the algebraic equations for each time-tick.

Note that the equations for subsequent time steps are coupled through terms obtained from numerical integration. So currents at $t = n + 1$ will be functions of currents at $t = n$. We will see this in more detail while constructing the companion models for various memory elements. Also, note that linear equations are recursively solved to traverse through simulation time by choosing a **time-step** (Δt). The value of this time step depends on the local truncation error (LTE). More details on the calculation of LTE can be found in [9]. The general idea is to maintain a sufficient tradeoff between simulation speed (by taking larger time steps) and accuracy (by taking smaller time steps).

Now we will learn how to solve the EMT problem from the viewpoint of the equivalent circuit approach (ECA). To obtain the time-domain response of the network with ECA, we apply **two (2) circuit-simulation tricks** to enable efficient construction of the linearized nodal equations ($Y_t V_t - J_t = 0$) for each time step (t) in the transient simulation.

First, rather than defining the ordinary differential equation (ODE) that captures the KCL constraint (sum of currents) at each node, we construct specialized symbolic stamps for each component in the network (e.g., resistance, inductance, etc.). Note that a stamp(s) here refers to term(s) that are to be added to the solution matrix. We use these stamps to implicitly construct the system matrix at each time tick. To do so, we parse through each circuit element in the network (for each time tick), and we add the stamps corresponding to this element in the solution matrix. Once completed, the matrix (completed by adding stamps from each component) will be equivalent to if we constructed the matrix by adding linearized nodal equations in each row. In this approach, rather than arduously constructing nodal equations for each node (imagine a circuit with million nodes), we only have to symbolically construct stamps for the number of types of circuit elements. Once that is done, all we need to do is add the stamps for each circuit element into the solution matrix.

Second circuit simulation trick pertains to simplifying the use of numerical integration of differential terms. Instead of applying numerical integration to all nodal ODE equations explicitly, we develop and use companion circuits for time derivative memory elements (e.g., inductance and capacitance) in the network. By applying these two tricks, the ECA approach, rather than tackling one nodal or loop equation at a time, builds modular circuits for various elements with embedded implicit symbolic linearization and discretization. This allows us to focus on a limited number of types of power grid elements (generator, load, etc.) instead of millions of nodal or loop equations.

In ECA, with the two tricks, we follow the following steps sequentially to enable modular construction and solution of the system matrix at each time tick without ever having to define the complete set of nodal equations explicitly:

- (1) Represent time-derivative elements in the circuit (e.g., inductor, capacitor etc.) with their companion circuits.
- (2) Develop a specialized stamp for each element in the circuit (with derivative elements replaced by their companion circuits).
- (3) $\forall t \in \tau = \{t_0, \dots, t_{final}\}$, parse through the overall network (generally a netlist or input file of sorts) and stamp the terms corresponding to each element in a sparse linear matrix (remember at this point, time-derivative terms are replaced by their companion models).
- (4) Solve the linear problem $Y_t V_t = J_t$, to obtain the solution of state variables at time t , and update and repeat until final time step.

This approach allows us to implicitly obtain and solve the linearized nodal equations without constructing them symbolically. This is highly beneficial for developing heuristics and for solving large-scale networks. More on that later.

1.1.1 Companion Model for Inductors and Trapezoidal Rule

Returning to the RL-load example, the inductor is the only element with a time derivative term in this network (see Figure 1.3). Therefore,

all we have to do is construct a companion model for the inductor. We then replace all the inductors in the network with their companion models. We now discuss the construction of these companion models.

This tutorial will use trapezoidal integration approximation to construct companion models. Other Euler-based or higher-order numerical integration methods can also be used following the same approach. Applying the trapezoidal integration approximation, the companion circuit for the inductor can be derived as follows (for more details, see [9]):

$$L \frac{di}{dt} = v \quad (1.1)$$

$$di = \frac{v dt}{L} \quad (1.2)$$

$$\int_{t_n}^{t_{n+1}} di = \int_{t_n}^{t_{n+1}} \frac{v dt}{L} \quad (1.3)$$

$$(i(t_{n+1}) - i(t_n)) = (v(t_{n+1}) + v(t_n)) \frac{\Delta t}{2L} \quad (1.4)$$

$$i(t_{n+1}) = i(t_n) + v(t_n) \frac{\Delta t}{2L} + v(t_{n+1}) \frac{\Delta t}{2L} \quad (1.5)$$

The equation in (1.5) has three terms. The last term $\frac{\Delta t}{2L}$ describes the relationship between current and voltage at time t_{n+1} , across the same branch, and is a conductance G . The first two terms are constants that are known from states at prior time-point t_n states and therefore are represented via a constant current source I . Aggregating these elements into a circuit, we get the equivalent circuit in Figure 1.4, and it is a companion model for an inductor with trapezoidal numerical integration approximation.

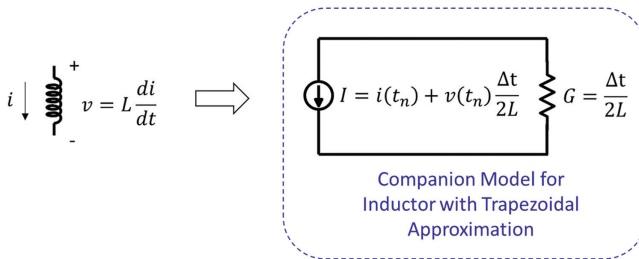


Figure 1.4: Inductor Trapezoidal Companion Circuit. Reconstructed from Pileggi, Carnegie Mellon ECE 18-762 notes [9].

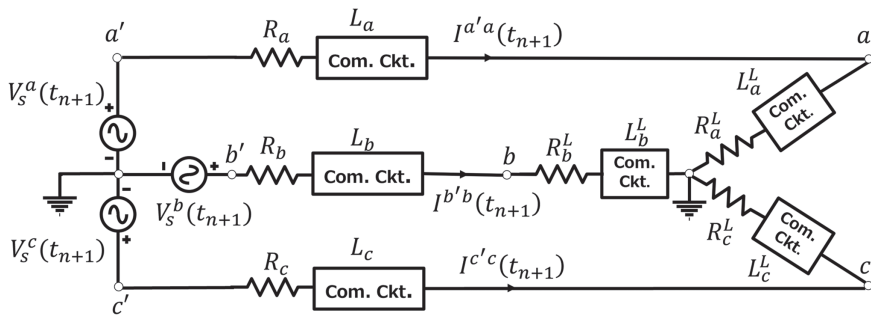


Figure 1.5: Equivalent circuit with companion circuits (com. ckt.).

With the companion model constructed, we can replace the inductor elements in Figure 1.3 with the respective companion circuits. Following this step, we obtain a time-dependent equivalent circuit (see Figure 1.5), which can be characterized entirely by linear algebraic equations. This set of equations is repeatedly solved to find the electromagnetic transient response of the network.

However, as discussed, the trick is to avoid explicitly constructing the complete equivalent circuit and corresponding nodal equations ($YV - J$) for each node. Instead, we will follow a modular approach. As we encounter each element in the netlist of the network, we will add the specialized stamps corresponding to that element into the sparse system matrix. In this example (see Figure 1.5), there are three ideal voltage sources, six resistance and six inductor models. Therefore, we will cover the stamps for these elements, which will be sufficient to construct the linearized nodal equations for each time-step ($Y_t V_t - J_t = 0$) for the network in Figure 1.5. By recursively solving these equations over time from the initial time-step t_0 to the final time step t_{final} , we can obtain the time-domain response of this network. Next, we will cover the stamps for these elements.

Note: We will refer to the sparse linear matrix for the system as Y from here on. Each time step will have a different system matrix notated by Y_t or sometimes simply by Y . Further, even for each time step, the matrix can be expressed as the sum of two separate matrices: $Y_t = Y_t^{lin} + Y_t^{nlin}$. Terms in Y_t^{nlin} change for each iteration of NR,

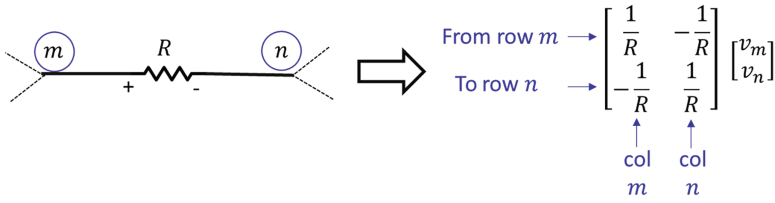


Figure 1.6: Resistance stamps in the nodal matrix. Redrawn from CMU, ECE, 18-762 notes [9].

whereas the Y_t^{lin} , remain fixed. Because this circuit is linear, we do not have Y_t^{nlin} .

1.1.2 Stamp for Resistors

The stamps for a resistance R between node m and n in the nodal matrix Y are shown in Figure 1.6. The general approach is that when you encounter resistance in an input file, add the terms defined in Figure 1.6 to sparse linear matrix Y in the index locations: (m, m) , (m, n) , (n, m) , and (n, n) . Here node m is the from node of the resistor, and n is the to node of the resistor.

1.1.3 Stamps for an Independent Current Source

An independent current source stamp I between node m and n will add no additional stamps to matrix Y but instead will add terms to column vector J as shown in Figure 1.7. The column vector J only consists of constant terms that are not a function of any state in the network. The negative sign in the J vector is due to moving the constant number to the RHS of the equation. However, keep in mind the direction of the current flow from the independent current source for the signage.

1.1.4 Stamps for Inductors

Inductor consists of time-derivative terms. Therefore, as a first step, each time we encounter an inductor in the netlist between node m and node n , we will have to stamp the corresponding companion circuit. Remember doing so will replace the time-derivative terms with algebraic expressions. The companion circuit consists of a current source and a

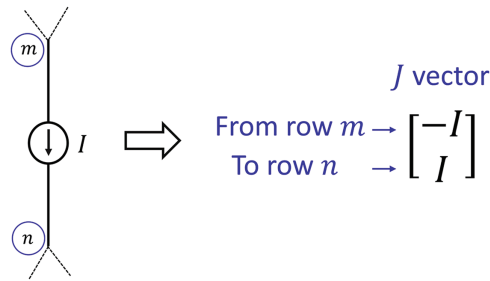


Figure 1.7: Independent current source stamps in J vector. Redrawn from Pileggi, CMU 18-762 notes [9].

resistor in parallel between nodes m and n as seen in Figure 1.4. As we have already covered stamps for resistance and current source, follow the steps in the previous subsections to stamp those components. The key thing to note is that stamp values for inductors in system matrix Y will change for each time step and must be updated accordingly. This is because the values of the independent current source in the companion circuit will be a function of currents and voltages from the last time tick t_n .

1.1.5 Stamps for Independent Voltage Source

Next, we derive the stamps for a voltage source. To include an independent voltage source between nodes m and n , nodal equations are augmented to include an additional row. With an extra row comes an additional variable representing the magnitude of current flow through the voltage source. The additional row (for each voltage source) constrains the potential difference between node m and node n :

$$V_m - V_n = V_S \quad (1.6)$$

and the additional current is notated by i . The stamps in the nodal matrix (Y) and source vector (J) for independent voltage source are shown in Figure 1.8.

Now that we have learned how to develop stamps for the resistor, inductor, current source, and voltage source, we can construct the nodal matrix (Y) and source vector (J) for the simple power system network

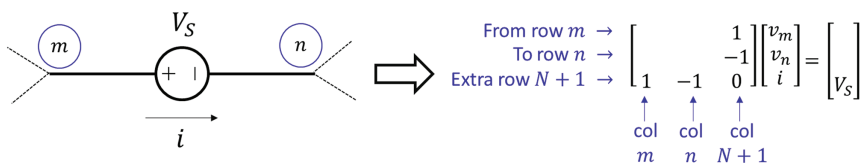


Figure 1.8: Independent voltage source stamps. Redrawn from Pileggi, CMU- 18-762 notes [9].

in Figure 1.3 for each time t_{n+1} . To obtain the time-domain response of the network, the set of linearized equations is recursively solved with a sparse matrix solver (using LU factorization). Note that to obtain the network's response at time $t = 0^+$, we require the network states at time $t_n = 0$. For this particular network, these are needed to initiate the terms in inductor stamps dependent on the prior time step. Not just that, it also helps dictate the system state at which the time-domain response begins.

1.1.6 Initialization

We can initialize the linear circuit through AC or DC analysis. In DC analysis, we replace the sinusoidal voltage sources with DC voltage sources. In AC analysis, we convert the inductors to an impedance at a single frequency.

Here we will describe the DC analysis approach to initialization. Let us assume the network is in the DC state between $t = -\infty$ and $t = 0$. At $t = 0$, it switches to the AC sinusoidal source, and let us assume the switching is smooth (i.e., DC voltage and AC voltage at time $t = 0$ are identical). To know the initial states of the circuit at $t = 0$, we perform DC analysis, wherein the following condition holds true for memory elements (inductors and capacitors):

$$L \frac{di}{dt} = 0 \quad @DC \quad (1.7)$$

$$C \frac{dv}{dt} = 0 \quad @DC \quad (1.8)$$

From these conditions in (1.7) and (1.8), we can infer that during the DC state, the inductor L^{DC} is short-circuited ($V_L = 0$) and the capacitor

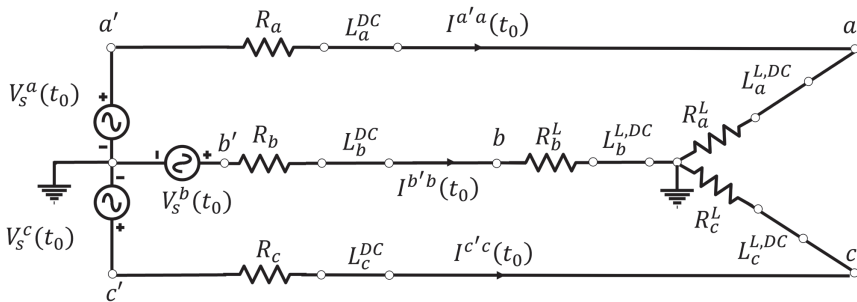


Figure 1.9: Corresponding equivalent circuit during DC state. Note that the sinusoidal voltage sources are set to a fixed DC value in the case of DC analysis.

C^{DC} is open-circuited ($I_C = 0$). For this example, the DC equivalent of the network in Figure 1.5 is shown via a network in Figure 1.9. With access to DC equivalent circuit, we stamp the elements in the circuit into the system matrix Y_{dc} and vector J_{dc} and solve them using a sparse linear solver. The solution is used to define the system state at $t = 0$.

1.1.7 Time-Domain Solution

With access to the initial state at $t = 0$, we recursively solve a system matrix (moving forward by Δt in each recursion) following steps described in Figure 1.10 to obtain the time-domain response from $t = 0$ to $t = t_{final}$. We can dynamically adjust Δt at each step based on the trade-off between simulation run-time and local truncation error (LTE). LTE is calculated by approximating the higher-order terms of the numerical integration. More details can be found in [9]. Note that dynamically adjusting LTE is not always necessary as for a purely linear system changing Δt will require re-generating the Δt dependent companion model terms at each iteration at an additional computational cost. Practically speaking, a happy medium can be obtained by periodically checking the LTE throughout the simulation and adjusting Δt if it is found high.

One facet that this example does not cover is how to handle nonlinear terms. We will learn how to include nonlinear terms in the following example with IM as an electric load. We will also learn to use an alternative mathematical form (other than MNA) to describe the

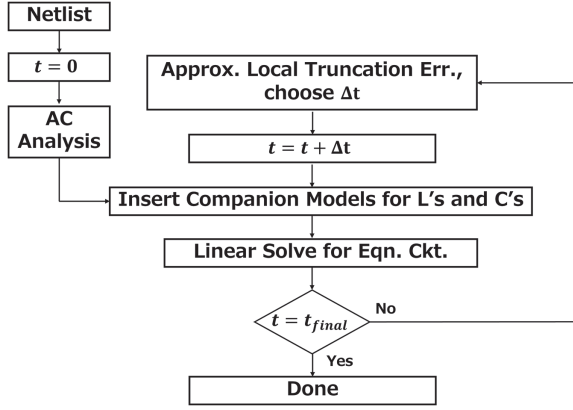


Figure 1.10: Running EMT simulation for linear networks.

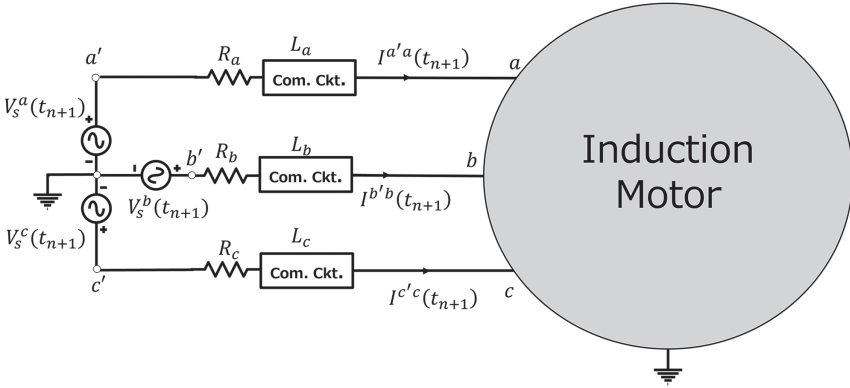


Figure 1.11: 2-bus network with induction motor load.

system's behavior. In addition, we will also learn how to incorporate non-electrical physics (i.e., IM's mechanical physics), like rotor speed, with the circuit-theoretic paradigm, into the mathematical formulation.

1.2 Induction Motor Load

In the second example, we will learn how to obtain the time-domain response for a non-linear grid component: the induction motor (IM). To construct a toy power grid network with IM, we will replace the RL load in the first example with an IM load (see Figure 1.11).

Analyzing IM in the abc frame can be difficult. The flux generated by the three-phase IM in abc frame has time-varying coefficients in its voltage terms due to the sinusoidal nature of the mutual inductance. This makes the analysis of three-phase IM cumbersome in the abc reference frame. However, this undesirable feature can be eliminated by using dq transformation. dq transformation is a linear transformation. It can be performed by choosing one of the three reference frames: (i) synchronous reference frame, (ii) stationary reference frame, and (iii) rotating reference frame. For more details on dq transformation, see [5].

For derivations in this work, we will use the stationary reference frame and a power invariant dq transformation. Doing so allows us to set $\theta = 0$ and simplify further calculations by having a time-invariant voltage source across the IM. In the stationary frame, abc variables (both currents and voltages) are converted to dq variables by following power invariant dq matrix transformation:

$$[F_{0dq}] = [P_\theta][F_{abc}] \quad (1.9)$$

where:

$$[P_\theta] = \sqrt{\frac{2}{3}} \begin{bmatrix} \sqrt{0.5} & \sqrt{0.5} & \sqrt{0.5} \\ \cos(\theta) & \cos(\theta - \lambda) & \cos(\theta + \lambda) \\ \sin(\theta) & \sin(\theta - \lambda) & \sin(\theta + \lambda) \end{bmatrix}, \theta = 0 \quad (1.10)$$

λ is the phase difference between phases abc during balanced operation:

$$\lambda = \frac{2\pi}{3} \text{rad} \quad (1.11)$$

After applying dq transformation to IM and decoupling non-IM abc components (see left of Figure 1.12) from dq components using controlled current and voltage sources, the network model in Figure 1.11 is modified to one in Figure 1.12.

Note that controlled current and voltage sources in Figure 1.12 encapsulate the math behind dq and inverse dq transformation for the variables. dq transformation is applied to capture abc network voltages and reflect those for IM sub-circuit in dq frame. Inverse dq transformation is used to convert the currents consumed by the IM in dq frame into abc frame (I_a^{IM} , I_b^{IM} , I_c^{IM}) such that they can be fed into rest of the network.

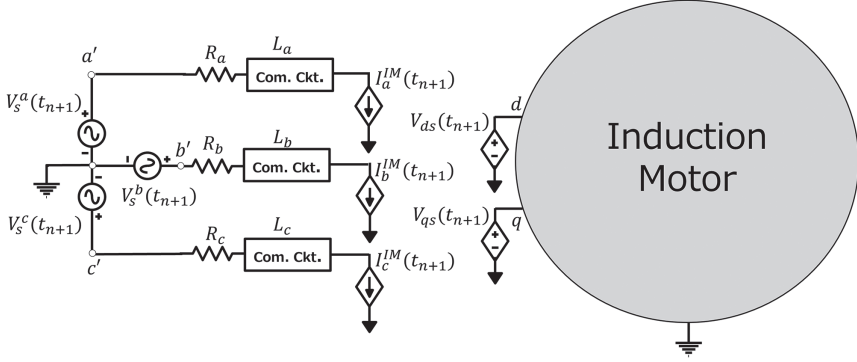


Figure 1.12: 2-bus network with induction motor load after dq transformation.

The transformed Figure 1.12 has three components: (i) the infinite bus represented by a set of wye-connected independent voltage sources, (ii) the transmission line represented by a series RL circuit, and (iii) an IM. We have already discussed the construction of stamps and companion circuits for infinite bus and transmission line elements (reiterating that in this tutorial the transmission line model is greatly simplified, see [3] for detailed models like pi model and Bergeron model). Now, we will learn how to represent the IM physics with an equivalent circuit. We will also learn how to replace the time derivative terms in the IM equivalent circuit with an analogous companion circuit.

The generalized equations that represent the electrical component of IM physics, independent of the reference frame choice in dq transformation, are expressed via the following KVL equations [5]:

$$V_{ds} = R_s I_{ds} + p\psi_{ds} + \psi_{qs}p\theta \quad (1.12)$$

$$V_{qs} = R_s I_{qs} + p\psi_{qs} - \psi_{ds}p\theta \quad (1.13)$$

$$V_{dr} = R_r I_{dr} + p\psi_{dr} + \psi_{qr}p\beta \quad (1.14)$$

$$V_{qr} = R_r I_{qr} + p\psi_{qr} - \psi_{dr}p\beta \quad (1.15)$$

where p is the differential operator and,

$$\psi_{ds} = (L_{ls} + L_m)I_{ds} + L_m I_{dr} \quad (1.16)$$

$$\psi_{dr} = (L_{lr} + L_m)I_{dr} + L_m I_{ds} \quad (1.17)$$

$$\psi_{qs} = (L_{ls} + L_m)I_{qs} + L_m I_{qr} \quad (1.18)$$

$$\psi_{qr} = (L_{lr} + L_m)I_{qr} + L_m I_{qs} \quad (1.19)$$

In the expressions above, L_{ls} and L_{lr} represent the leakage inductance of the stator circuit and rotor circuit, respectively. L_m is the mutual inductance between the rotor and stator circuits. R_s and R_r are the stator and rotor resistance, respectively. ψ represents the flux component across each electrical sub-circuit. The nonlinearity in the electrical part of the IM is due to the speed voltage terms ($\psi_{qr}\omega_r, \psi_{dr}\omega_r$).

In addition to the IM electrical part equations in (1.12)–(1.19), the mechanical IM part is represented by a single differential equation, the swing equation (1.20):

$$p\omega_r = \frac{(T_E - T_L - D\omega_r)}{J} \quad (1.20)$$

where, electrical torque T_E is further described by interactions between the IM currents in the electrical sub-circuit, which also introduces nonlinearities in the IM model:

$$T_E = \frac{3}{4}L_m N_p (I_{dr}I_{qs} - I_{qr}I_{ds}) \quad (1.21)$$

T_E , the electrical torque, is given in $N.m$ and J , the motor net inertia, in $\text{kg} \cdot \text{m}^2$. N_p is the number of poles in the IM. The load torque T_L is generally represented as a polynomial function of rotor speed.

We can further simplify the electrical components' expressions because we use the stationary reference frame for dq transformation. In the stationary reference, $\theta = 0$, $\beta = -\theta_r$ and $p\beta = -\omega_r$. Furthermore, $p\theta = \omega_s = 0$. With these simplifications and reducing the expressions in (1.12)–(1.19), the following form for the IM's electrical part, in the

stationary frame, can be obtained:

$$V_{ds} - R_s I_{ds} - L_s p I_{ds} - L_m p I_{dr} = 0 \quad (1.22)$$

$$V_{qs} - R_s I_{qs} - L_s p I_{qs} - L_m p I_{qr} = 0 \quad (1.23)$$

$$V_{dr} - R_r I_{dr} - L_r p I_{dr} - L_m p I_{ds} + \omega_r L_r I_{qs} + \omega_r L_m I_{qr} = 0 \quad (1.24)$$

$$V_{qr} - R_r I_{qr} - L_r p I_{qr} - L_m p I_{qs} - \omega_r L_r I_{dr} - \omega_r L_m I_{ds} = 0 \quad (1.25)$$

where,

$$L_s = L_{ls} + L_m \quad (1.26)$$

$$L_r = L_{lr} + L_m \quad (1.27)$$

$$V_{dr} = 0 \quad (1.28)$$

$$V_{qr} = 0 \quad (1.29)$$

Note that the rotor windings are shorted in most IM designs, and hence voltages V_{dr} and V_{qr} are set to 0.

Now to construct the equivalent circuit for the IM model, we map the four KVL expressions in (1.22)–(1.25) and one KCL expression in (1.20) to Figure 1.13.

One can observe in Figure 1.13 that aside from self-inductance (L_s and L_r), the IM equivalent circuit also includes time-derivative terms (given by p or \dot{I}/ω_r) for mutual inductance (L_m) and rotor speed (ω_r). We have previously learned how to approximate time-derivative terms for self-inductance by constructing and stamping the corresponding companion circuits. Here, we will derive the companion circuits for combined self and mutual inductance elements (see blue elements in Figure 1.14). We will use a simple two-coil example in Figure 1.15 to develop the companion circuit for combined self- and mutual inductance. Later in the section, we will learn how to construct the companion circuit for time-derivative terms corresponding to rotor speed. We will finish by discussing how the nonlinear elements in the IM equivalent circuit are handled and subsequently added to the system matrix.

We construct the companion circuit for combined self and mutual inductances to approximate its time-derivative behavior by a set of algebraic equations. This way, we can approximately solve a set of

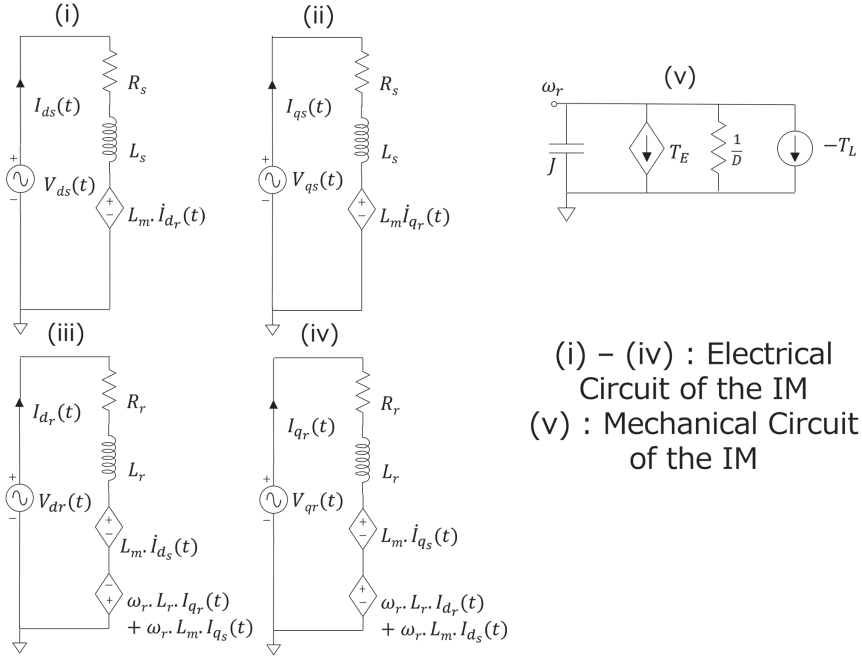


Figure 1.13: Equivalent circuit for IM in the stationary frame.

ODEs parameterized by self- and mutual-inductances by recursively solving a set of algebraic equations over time. We begin the derivation by observing a two-coil example in Figure 1.15, which has two pairs of self- and mutual-inductance (self: L_{11} or L_{22} , mutual: L_{12}). We focus on voltage V_1 expression in the first coil. It can be represented as a sum of the voltage across the self and mutual inductances:

$$V_1(t) = L_{11} \frac{dI_1}{dt} + L_{12} \frac{dI_2}{dt} \quad (1.30)$$

The voltage expression in (1.30) is an ODE that includes time-derivative terms. Therefore, we apply the trapezoidal integral rule to approximate the ODE's solution. With the trapezoidal integral rule applied, the voltage across self- and mutual-inductance (1.30) can be represented by

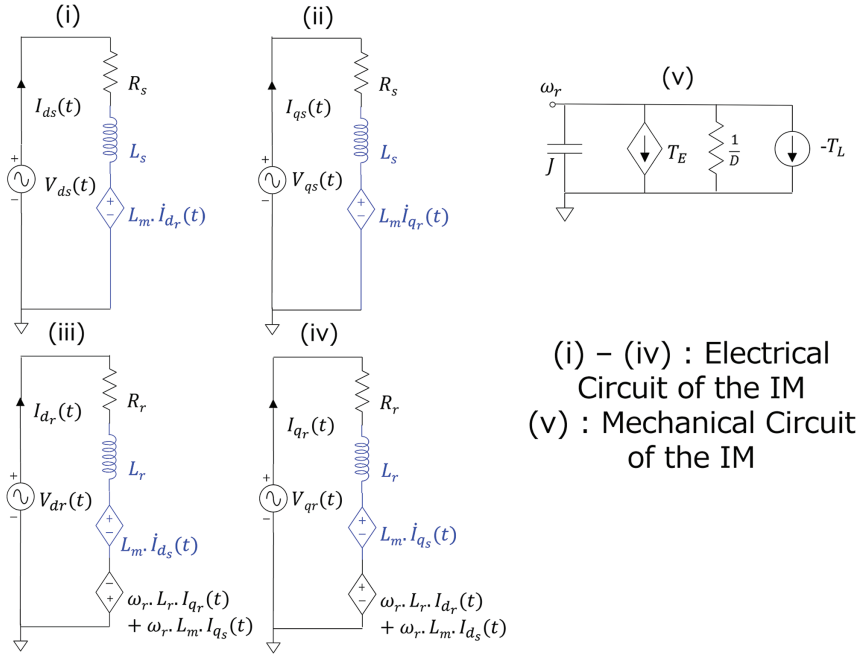


Figure 1.14: Illustration of combined self and mutual inductances in IM.

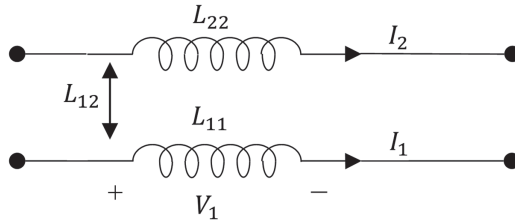


Figure 1.15: Two coil example.

the sum of the following difference equations, which are purely algebraic terms:

$$V_{11}(t_{n+1}) = \frac{2L_1}{\Delta t} (I_1(t_{n+1}) - I_1(t_n)) - V_{11}(t_n) \quad (1.31)$$

$$V_{12}(t_{n+1}) = \frac{2L_{12}}{\Delta t} (I_2(t_{n+1}) - I_2(t_n)) - V_{12}(t_n) \quad (1.32)$$

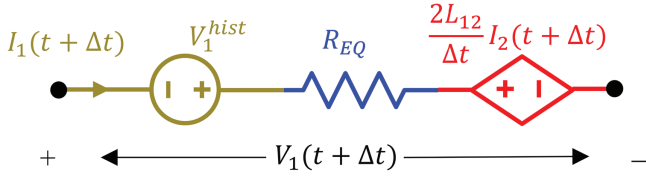


Figure 1.16: Two coil example companion circuit.

Therefore the total voltage induced in coil 1 ($V_1(t) = V_{11}(t) + V_{12}(t)$) is given by:

$$V_1(t_{n+1}) = \frac{2L_1}{\Delta t} (I_1(t_{n+1}) - I_1(t_n)) - V_{11}(t_n) + \frac{2L_{12}}{\Delta t} (I_2(t_{n+1}) - I_2(t_n)) - V_{12}(t_n) \quad (1.33)$$

and after re-arranging the terms:

$$V_1(t_{n+1}) = \frac{2L_1}{\Delta t} I_1(t_{n+1}) + \frac{2L_{12}}{\Delta t} I_2(t_{n+1}) - \left(\frac{2L_1}{\Delta t} I_1(t_n) + \frac{2L_{12}}{\Delta t} I_2(t_n) + V_{12}(t_n) + V_{11}(t_n) \right) \quad (1.34)$$

The voltage at time t_{n+1} across the coil 1 is approximated using (1.34) and is represented as an equivalent circuit in Figure 1.16. The third term is only dependent on the historical values of the variables and can be represented by an independent voltage source V_1^{hist} . Similarly, a resistance R_{EQ} represents first term as it maps the linear relationship between the voltage and current across the same branch and a current controlled voltage source represents the second term, as it maps the voltage contribution due to current in the second coil.

By replacing the time-derivative terms for self- and mutual-inductance with the corresponding companion circuits in Figure 1.16, we can solve the set of ODEs corresponding to the electrical part of IM, recursively over time. But before, we must learn how to address the time-derivative term in the mechanical part of the IM model. The variables in the mechanical part impact the electrical part and vice-versa. Therefore we need to solve the mechanical part concurrently with the electrical portion of IM. The approach for solving the mechanical part is trivial. On close observation, one can see the mechanical circuit of the IM is analogous to a parallel RC electrical circuit with current sources, where inertia J is analogous to capacitance C , T_L is analogous to a constant current source I , D is analogous to a conductance G , T_E is analogous

to a current controlled current source (see (1.21)), and ω_r is analogous to the voltage at the mechanical circuit node. The only time-derivative term in the mechanical circuit is the partial derivative of rotor speed times inertia ($J \frac{d\omega_r}{dt}$), which is analogous to a current through a capacitor. Therefore, we will learn how to construct a capacitor's companion circuit to replace the time-derivative term in the mechanical part of IM. By replacing the time-derivative term, we can represent the physics of IM's mechanical part with a single nodal equation that can be solved recursively over time (assuming we have dealt with nonlinearities).

To construct the companion model for a capacitor, we will use trapezoidal integration approximation following the same approach we used for inductors. Applying the trapezoidal integration approximation, the expression for current/voltage through/across a capacitor at time t_{n+1} given values at time t_n can be expressed as follows:

$$C \frac{dv}{dt} = i \quad (1.35)$$

$$dv = \frac{idt}{C} \quad (1.36)$$

$$\int_{t_n}^{t_{n+1}} dv = \int_{t_n}^{t_{n+1}} \frac{idt}{C} \quad (1.37)$$

$$(v(t_{n+1}) - v(t_n)) = (i(t_{n+1}) + i(t_n)) \frac{\Delta t}{2C} \quad (1.38)$$

$$v(t_{n+1}) = v(t_n) + i(t_n) \frac{\Delta t}{2C} + i(t_{n+1}) \frac{\Delta t}{2C} \quad (1.39)$$

The equation in (1.39) has three terms. The last term $\frac{\Delta t}{2C}$ describes the relationship between current and voltage at time t_{n+1} and is a conductance G . The first two terms are constants that are known from prior time-point t_n and therefore are represented via a constant voltage source V . Aggregating these elements in series, we get the equivalent circuit in Figure 1.17, representing a capacitor's companion model with trapezoidal numerical integration approximation. For representing the speed time-derivative terms in IM, we will use analogous symbols in the companion model: inertia J for capacitor C and rotor speed ω_r for voltage v .

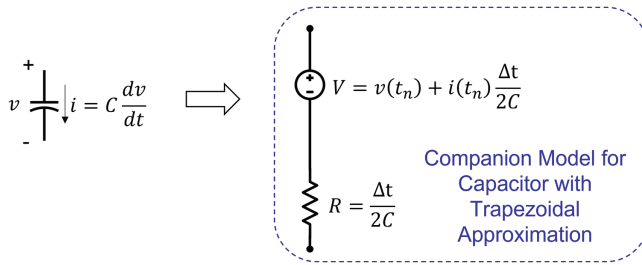


Figure 1.17: Capacitor trapezoidal companion circuit. Redrawn from Pileggi, 18-762 notes [9].

1.2.1 Handling Nonlinear Terms

In the case of IM, we began with a set of models characterized by differential-algebraic equations (DAEs), which had both differential and nonlinear terms. As a first step, we built and used companion circuits to approximate the differential term behavior with algebraic terms. Next, we linearize the nonlinear terms and solve them iteratively to approximate the nonlinear behavior.

We linearize the nonlinear terms with first-order Taylor's approximation, like in nonlinear power flow analysis. However, instead of linearizing complete nodal or loop equations (e.g., KCL or KVL), we linearize the nonlinear terms corresponding to each component modularly by developing linearized equivalent circuits. Then for each iteration of NR, we iteratively update the stamps corresponding to these components in the system matrix Y and resolve.

Let us write out and understand the general form and application of first-order Taylor approximation. For general nonlinear function $f(x)$ ($x: \mathbb{R}^{N \times 1}$), the first order Taylor approximation is given by:

$$f^{k+1}(x) \approx f^k(x) + f'(x)^T (x^{k+1} - x^k) \quad (1.40)$$

With nonlinear $f(x)$ linearized around x_k from the k^{th} iteration, the next iterate can be obtained via a linear solve. We will use the general theory now to understand the linearization of IM terms.

In the IM model in Figure 1.13, for the electrical part, the non-linearity stems from four speed-voltage terms ($\omega_r L_r I_{dr}$, $\omega_r L_m I_{ds}$, $\omega_r L_r I_{qr}$, and $\omega_r L_m I_{qs}$) in the dr and qr sub-circuits (iii, and iv in Figure 1.13).

There are additional nonlinearities in the mechanical sub-circuit (v in Figure 1.13)

To further explore how to add IM nonlinear terms to the Y matrix, let us consider the nonlinear terms in the KVL expression (1.25) for subcircuit (iv) in Figure 1.13:

$$f_{qr}^{NL}(\omega_r, I_{ds}, I_{dr}) = -\omega_r L_r I_{dr} - \omega_r L_m I_{ds} \quad (1.41)$$

(1.41) is a nonlinear function of the rotor speed (ω_r), direct-axis rotor current (I_{dr}), and direct-axis stator current (I_{ds}). The linearized approximation for this expression for the $(k+1)^{\text{th}}$ NR iteration is:

$$\begin{aligned} f_{qr}^{k+1}(\omega_r, I_{dr}, I_{ds}) &= f_{qr}^k(\omega_r, I_{dr}, I_{ds}) + \left(\frac{\partial f}{\partial \omega_r} \right)_k (\omega_r^{k+1} - \omega_r^k) + \left(\frac{\partial f}{\partial I_{dr}} \right)_k (I_{dr}^{k+1} - I_{dr}^k) \\ &\quad + \left(\frac{\partial f}{\partial I_{ds}} \right)_k (I_{ds}^{k+1} - I_{ds}^k) \end{aligned} \quad (1.42)$$

where:

$$f^k(\omega_r, I_{dr}, I_{ds}) = -\omega_r^k L_r I_{dr}^k - \omega_r^k L_m I_{ds}^k \quad (1.43)$$

$$\left(\frac{\partial f}{\partial \omega_r} \right)_k = -L_r I_{dr}^k - L_m I_{ds}^k \quad (1.44)$$

$$\left(\frac{\partial f}{\partial I_{dr}} \right)_k = -\omega_r^k L_r \quad (1.45)$$

$$\left(\frac{\partial f}{\partial I_{ds}} \right)_k = -\omega_r^k L_m \quad (1.46)$$

Similar to the treatment of nonlinear terms in (1.25), other nonlinear terms in the IM are also linearized and stored in symbolic form. These terms are added (or stamped) in the system matrix Y following the discussion in next subsection and updated for each iteration of NR.

1.2.2 Stamping Controlled Current Sources

In the IM linearized equivalent circuit, we end up with two new circuit elements, a current-controlled voltage source (CCVS) and a current-controlled current source (CCCS). The CCVS originate from linearizing speed current terms in (1.22)–(1.25) (see terms in (1.42) for F_{qr} KVL constraint). The CCCS terms originate while linearizing the T_E equation. We briefly describe how those terms are added to the system matrix.

$$\text{Row for } F_{qr} \text{ KVL} \rightarrow \begin{bmatrix} X & X & X & X & X \\ X & X & X & X & X \\ X & X & X & X & X \\ \frac{\partial F_{qr}}{\partial I_{ds}} & X & \frac{\partial F_{qr}}{\partial I_{dr}} & X & X \\ X & X & X & X & X \end{bmatrix} \begin{bmatrix} I_{ds} \\ I_{qs} \\ I_{dr} \\ I_{qr} \\ \omega_r \end{bmatrix}$$

Figure 1.18: Off-diagonal entries for modeling CCVS in IM's F_{qr} electrical sub-circuit.

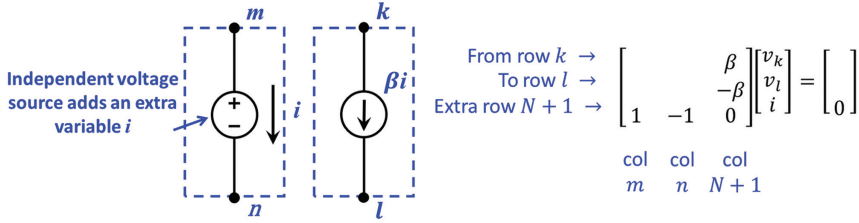


Figure 1.19: Stamps for CCCS. Reconstructed from Pileggi, Carnegie Mellon ECE 18-762 notes [9].

As we use KVL to describe the IM's electrical equations, the current-controlled voltage sources only show up as off-diagonal terms in the system matrix. For example, the CCVS stamps for F_{qr} KVL expression are shown in the system matrix in Figure 1.18. These do not include the rotor speed dependent terms.

Next, we discuss CCCS stamps, which are in the mechanical equation and are stamped following the MNA approach in Example 1. The stamps for a generic CCCS between nodes k and l are shown in Figure 1.19. In general, to measure the independent current on which the controlled source depends, we add a zero-valued voltage source (ammeter) between nodes m and n . Note that adding a voltage source results in an addition of a row to the Y -matrix, and in return, we get the current value through the source. Now, as the dependent current source between nodes k and l is valued β times the measured current i , for the example in Figure 1.19, we add corresponding terms to KCLs for nodes m and n . However, for the IM example specifically, we do not need an additional zero-valued voltage source as IM currents (I_{qr} , I_{qs} , I_{dr} , and I_{ds}) are already variables and easily accessible.

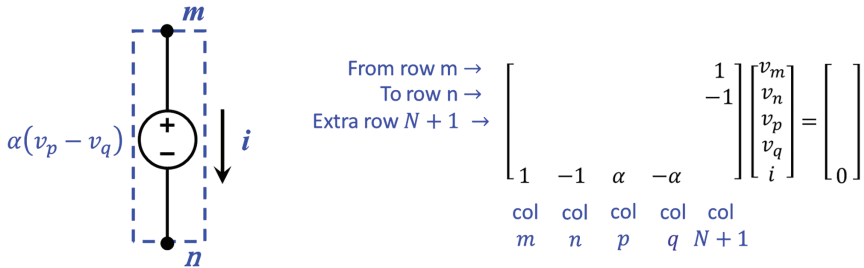


Figure 1.20: Stamps for VCVS. Reconstructed from Pileggi, Carnegie Mellon ECE 18-762 notes [9].

The IM in dq frame is coupled to the rest of the circuit with controlled sources as well. These are given by I_a^{IM} , I_b^{IM} , I_c^{IM} and V_{ds} , V_{qs} in Figure 1.12. The first set I_a^{IM} , I_b^{IM} , I_c^{IM} , which map the currents consumed by the IM in dq frame into abc frame, are modeled via CCCS. The underlying math defining the relationship between currents in dq and abc frames is given by inverse dq transformation. The stamps for these sources are covered in the CCCS description above. The second set, which maps the abc voltages to dq voltages with linear dq transformation, is represented via voltage-controlled voltage sources (VCVS). The stamps for generic VCVS are shown in Figure 1.20. Here, the voltage source across nodes m and n equals α times the difference between voltages at nodes p and q . This relationship is captured in the extra node $N + 1$ added to the matrix, which also gives us the current i through the voltage source. The current is added to KCL rows for nodes m and n . In the IM, the voltage V_{ds} and V_{qs} will be functions of V_a^{IM} , V_b^{IM} and V_c^{IM} .

1.2.3 Stamping IM Equations

So far, we discussed addressing nonlinear and differential terms modularly for each component. We also learned how individual components are added to the system matrix. The next step is to learn how the various terms come together in the overall Y matrix to satisfy system-level physics for the network in Figure 1.12. We must also learn the updation of these terms for each iteration (for nonlinear elements) and

each recursion (for each time step forward) to obtain a time-domain solution for the overall circuit.

Let us begin with non-IM linear components in Figure 1.12. We add them into the system matrix following the MNA approach described in Example 1 of this tutorial. We update the companion circuit terms for memory elements every instance we move forward in time. For the IM terms, we will use an alternative approach. If we were to stamp the IM circuit elements using MNA, we would end up with ~ 20 additional nodes and corresponding equations for each IM we encounter in the network (refer to Figure 1.13). Therefore, to reduce the dimension of IM equations, in this tutorial, we instead used a combination of Kirchhoff Voltage Law (KVL)-based loop constraints and KCL-based nodal constraints to stamp the IM model's equivalent circuit elements. With this approach, we will stamp the equations for electrical sub-circuits (i through iv in Figure 1.13) in 4 new rows of the system matrix Y following KVL-based loop constraints. Specifically, for each electrical sub-circuit in Figure 1.13 and (1.22)–(1.25), moving from left to right, we will add stamps (for voltages) for each element such that the net sum of voltages in the loop is equal to zero. For the mechanical part (no. (v) in Figure 1.13), we will add terms to matrix Y (including linearized T_E and companion circuits $\frac{\partial \omega_r}{\partial t}$) following KCL-based nodal constraint. With this approach, we only add five new variables and corresponding rows for each instance of IM (instead of ~ 20). The variables are I_{ds} , I_{qs} , I_{dr} , I_{qr} , and ω_r .

Recall that before stamping the terms into the system matrix Y , we have to perform two steps. First, we will replace the circuit elements with time-derivative terms with corresponding equivalent companion circuits. Second, we will replace the circuits with nonlinear terms with their linearized approximation as shown for F_{qr} in (1.42). For instance, the nonlinear symbolic stamps for F_{qr} loop constraint in (1.25) is added to the Y matrix as shown in Figure 1.21.

Next, to obtain the transient response of the nonlinear IM network, we will recursively solve the system matrix Y over time, and we will perform iterations to solve the nonlinear terms. We will update the linearized terms in each iteration. We will update the terms in companion circuits only when we recursively move forward in time.

$$\text{Row for } F_{qr} \text{ KVL} \rightarrow \begin{bmatrix} X & X & X & X & X \\ X & X & X & X & X \\ X & X & X & X & X \\ \frac{\partial F_{qr}}{\partial I_{ds}} & X & \frac{\partial F_{qr}}{\partial I_{dr}} & X & \frac{\partial F_{qr}}{\partial \omega_r} \\ X & X & X & X & X \end{bmatrix} \begin{bmatrix} I_{ds} \\ I_{qs} \\ I_{dr} \\ I_{qr} \\ \omega_r \end{bmatrix} = \begin{bmatrix} X \\ X \\ X \\ F_{qr}^{hist} \\ X \end{bmatrix}$$

Figure 1.21: Nonlinear IM stamps for F_{qr} KVL equation in (1.25).

1.2.4 Initialization

Next, we discuss how to initialize the IM network. In general, many approaches are available to initialize the network and we will discuss one such approach.

Remember because of dq-transformation on IM variables, two sub-circuits evolve (see left and right of Figure 1.12). The sub-circuit on the left without the IM equations is a linear circuit. AC analysis can be used to obtain the initial condition for these, assuming a rated complex current draw by IM (which models the I_a , I_b , and I_c in Figure 1.12). A good guess for a complex current draw by IM can be obtained by running power flow with IM modeled as a PQ load and calculating the current from the solution ($I = S_{IM}^*/V_{IM}^*$). Initialization of the IM components requires handling the nonlinearities due to speed-flux terms. With the proper choice of the reference frame in dq-transformation, time-invariant voltage sources (V_{ds} and V_{qs} are constants) across the IM can be obtained. We can then use DC analysis to obtain the initial conditions for the IM circuit. The V_{ds} , and V_{qs} voltages are DC-values in the rotating reference frame. We can short the inductor and open the capacitors to obtain the steady-state initial conditions with the source voltages as DC values. The rated voltage at the IM terminals for dq-transformation can be obtained from the power flow solution V_{IM} . In reality, with a slightly more involved procedure, we can get exact initial conditions for the IM circuit by solving a set of nonlinear equations representing the overall circuit iteratively using NR.

Generally, one must note that initializing large complex EMT networks is not trivial, and many commercial tool manuals [10], book chapters [9], and research papers are devoted to the study of efficiently initializing the network. In power systems, a common practice is to run

load flow and then map the load flow solution from the positive sequence frequency domain to three time-domain to initialize large power grid EMT networks. However, with emerging inverter-based resources, this approach may no longer work.

1.2.5 Nonlinear Time-Domain Simulation

For the nonlinear IM problem, with access to the initial state at $t = 0$, we repeatedly solve a system matrix Y (moving forward by Δt in each recursion) following steps described in Figure 1.22 to obtain the time-domain response from $t = 0$ to $t = t_{final}$. How we update the system matrix in the nonlinear analysis is different than in the case of linear analysis in Section 1.1.7. Linearized terms in the system matrix Y corresponding to nonlinear devices are updated *iteratively* after each linear solve. The memory-stamps corresponding to the devices with differential terms are only updated each time we move forward in time (*recursion*). We only move forward in time once the nonlinear stamps have converged (i.e., no considerable difference in the value of linearized stamps in subsequent iterations). As in the linear case, we dynamically adjust Δt at each step based on the trade-off between simulation run-time and local truncation error (LTE).

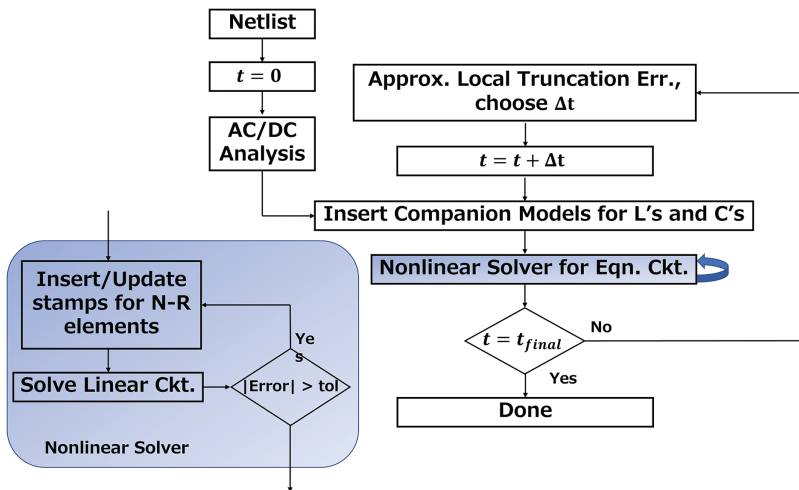


Figure 1.22: Running EMT simulation for nonlinear networks.

Conclusion and Resources for a Deeper Dive

This monograph covers the fundamentals of building power systems EMT tools using circuit theory with the aid of two simple examples. The tutorial is introductory, and it does not include many critical topics in detail, such as solver initialization, circuit-simulation heuristics for steep nonlinear models, sparse matrices, new power electronics-based models, and model reduction. For a deeper dive into the subject of EMT simulations, I recommend the following references [3], [2], [9], [10], and [6].

Parameter values for the two examples and a few others are included in the .json format and shared in the following publicly available git repo: https://github.com/amritanshup7/Tutorial-Circuit-based-Electromagnetic-Transient-Simulation/blob/main/example2_IM/IM_circuit.json. Anyone interested in EMT tool implementation for the two examples in Python language can reach out to me at amritanshu.pandey@uvm.edu.

Notation

Notations for parameters and symbols

Parameters	Description
Simulation Variables and Parameters.	
Y	Nodal admittance matrix
V	Solution vector of system states
J	Independent source vector
k	iteration count
Δt	time-step
t	time
DQ Transformation	
θ	Arbitrary value for stationary, synchronous, or rotating reference frame (rad)
β	Rotor angular position wrt to reference frame (rad)
λ	$2\pi/3$ (rad)
Linear Circuit Example	
R_a, R_b, R_c	Line resistances for phases abc (Ω)
L_a, L_b, L_c	Line inductances for phases abc (H)
R_a^L, R_b^L, R_c^L	Load resistances for phases abc (Ω)
L_a^L, L_b^L, L_c^L	Load inductances for phases abc (H)
Induction Motor Example	
F_{dr}	Nonlinear KVL equation for IM's dr subcircuit
F_{ds}	Nonlinear KVL equation for IM's ds subcircuit
F_{qr}	Nonlinear KVL equation for IM's qr subcircuit
F_{qs}	Nonlinear KVL equation for IM's qs subcircuit
R_s	Rotor resistance (Ω)
R_s	Stator resistance (Ω)
L_s	Stator self-inductance (H)
L_r	Rotor self-inductance (H)
L_m	Magnetizing inductance (H)

Parameters	Description
L_{ls}	Stator leakage inductance (H)
L_{lr}	Rotor leakage inductance (H)
ω_s	Synchronous angular velocity (rad/s)
ω_r	Angular velocity of the rotor (rad/s)
I_{dr}	Current in rotor's direct axis (A)
I_{ds}	Current in stator's direct axis (A)
I_{qr}	Current in rotor's quadrature axis (A)
I_{qs}	Current in stator's quadrature axis (A)
V_{dr}	Voltage across rotor's direct axis (V)
V_{ds}	Voltage across stator's direct axis (V)
V_{qr}	Voltage across rotor's quadrature axis (V)
V_{qs}	Voltage across stator's quadrature axis (V)
ψ_{dr}	Flux linkages across rotor's direct axis (Weber)
ψ_{ds}	Flux linkages across stator's direct axis (Weber)
ψ_{qr}	Flux linkages across rotor's quadrature axis (Weber)
ψ_{qs}	Flux linkages across stator's quadrature axis (Weber)
J	Combined inertia of motor and load (kgm^2)
D	Combined viscous friction of motor and load ($\text{N} \cdot \text{m}/(\text{rad/s})$)
θ_r	Motor mechanical angular position (rad)
T_L	Motor load torque (Nm)
T_E	Motor electrical torque (Nm)
N_p	number of poles

Acknowledgments

Tim McNamara and Naeem Turner-Bandele have both TA'ed the course where this material was taught and they have spent a significant amount of time ensuring that the derivations and explanations in this monograph are precise and accurate. Tim also provided valuable feedback on the structure and content. I also want to acknowledge my Ph.D. advisor Larry Pileggi whose book, lecture slides, and guidance were critical in compiling this monograph.

References

- [1] P. J. Davis and P. Rabinowitz, *Methods of Numerical Integration*. Courier Corporation, 2007.
- [2] H. W. Dommel, “Digital computer solution of electromagnetic transients in single-and multiphase networks,” *IEEE Transactions on Power Apparatus and Systems*, no. 4, pp. 388–399, 1969. DOI: [10.1109/TPAS.1969.292459](https://doi.org/10.1109/TPAS.1969.292459).
- [3] H. W. Dommel, *EMTP Theory Book*. Microtran Power System Analysis Corporation, 1996. DOI: [10.1137/1.9780898718898](https://doi.org/10.1137/1.9780898718898).
- [4] C. T. Kelley, *Solving Nonlinear Equations with Newton’s Method*. SIAM, 2003.
- [5] R. J. Lee, P. Pillay, and R. G. Harley, “D, q reference frames for the simulation of induction motors,” *Electric Power Systems Research*, vol. 8, no. 1, pp. 15–26, 1984. DOI: [10.1016/0378-7796\(84\)90030-0](https://doi.org/10.1016/0378-7796(84)90030-0).
- [6] F. Milano, *Power System Modelling and Scripting*. Springer Science and Business Media, 2010. DOI: [10.1007/978-3-642-13669-6](https://doi.org/10.1007/978-3-642-13669-6).
- [7] A. Pandey, M. Jereminov, X. Li, G. Hug, and L. Pileggi, “Unified power system analyses and models using equivalent circuit formulation,” in *2016 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, IEEE, 2016. DOI: [10.1109/ISGT.2016.7781182](https://doi.org/10.1109/ISGT.2016.7781182).

- [8] A. Pandey, M. Jereminov, M. R. Wagner, D. M. Bromberg, G. Hug, and L. Pileggi, “Robust power flow and three-phase power flow analyses,” *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 616–626, 2018. DOI: [10.1109/TPWRS.2018.2863042](https://doi.org/10.1109/TPWRS.2018.2863042).
- [9] L. Pillage, *Electronic Circuit & System Simulation Methods (SRE)*. McGraw-Hill, Inc., 1998. DOI: [10.1109/MCD.1998.735799](https://doi.org/10.1109/MCD.1998.735799).
- [10] PSCAD, “Pscad initialization tool—revision 1, written for pscad 5.0,” November 24, 2021.